

Appl. No. 10/605,716  
Amdt. dated January 05, 2006  
Reply to Office action of November 07, 2005

**Amendments to the Claims:**

1. (currently amended) A processor comprising:  
a ~~central processing unit (CPU) processing data according to an instruction set;~~  
a data memory for storing non-stack data;  
5 a stack memory for storing stack data, where the stack memory is different from the  
data memory;  
a memory address generator coupled to the data memory for producing addresses  
to for accessing the data memory; and  
a stack pointer generator coupled to the stack memory for producing a stack pointers  
10 for accessing to access the stack memory; and  
a central processing unit (CPU) coupled to the memory address generator and the  
stack pointer generator, the central processing unit for processing non-stack data and  
stack data according to an instruction set;  
wherein the stack pointer generator is further for producing a software stack pointer  
15 to access the stack memory when passing parameters to subroutines of the  
central processing unit.
2. (currently amended) The processor of claim 1 wherein the processor is a ~~Micro-Computer System (MCS)-series processor~~microcontroller.
- 20 3. (original) The processor of claim 1 wherein the processor processes an 8-bit instruction set.
4. (original) The processor of claim 3 wherein the data memory is 256 bytes.
- 25 5. (original) The processor of claim 3 wherein the stack memory is 256 bytes.
6. (new) The processor of claim 1 wherein the stack pointer generator is further for incrementally increasing the stack pointer to point to a next address when used by

Appl. No. 10/605,716  
Amdt. dated January 05, 2006  
Reply to Office action of November 07, 2005

the central processing unit, and for decreasing the software stack pointer from a predetermined starting position when passing parameters to subroutines of the central processing unit.

- 5     7. (new) A method for providing a processor with unshared stack memory, the method comprising:
- providing a data memory for storing non-stack data;
- providing a stack memory for storing stack data;
- producing addresses to access the data memory;
- 10     producing a stack pointer for accessing the stack memory;
- providing a central processing unit (CPU) for processing non-stack data and stack data according to an instruction set; and
- producing a software stack pointer to access the stack memory when passing parameters to subroutines of the central processing unit.
- 15
8. (new) The method of claim 7 wherein the processor is a microcontroller.
9. (new) The method of claim 7 further comprising the processor processing an 8-bit instruction set.
- 20
10. (new) The method of claim 9 further comprising the data memory being 256 bytes.
11. (new) The method of claim 9 further comprising the stack memory being 256 bytes.
- 25     12. (new) The method of claim 7 further comprising increasing the stack pointer incrementally to point to a next address when used by the central processing unit, and decreasing the software stack pointer from a predetermined starting position when passing parameters to subroutines of the central processing unit.